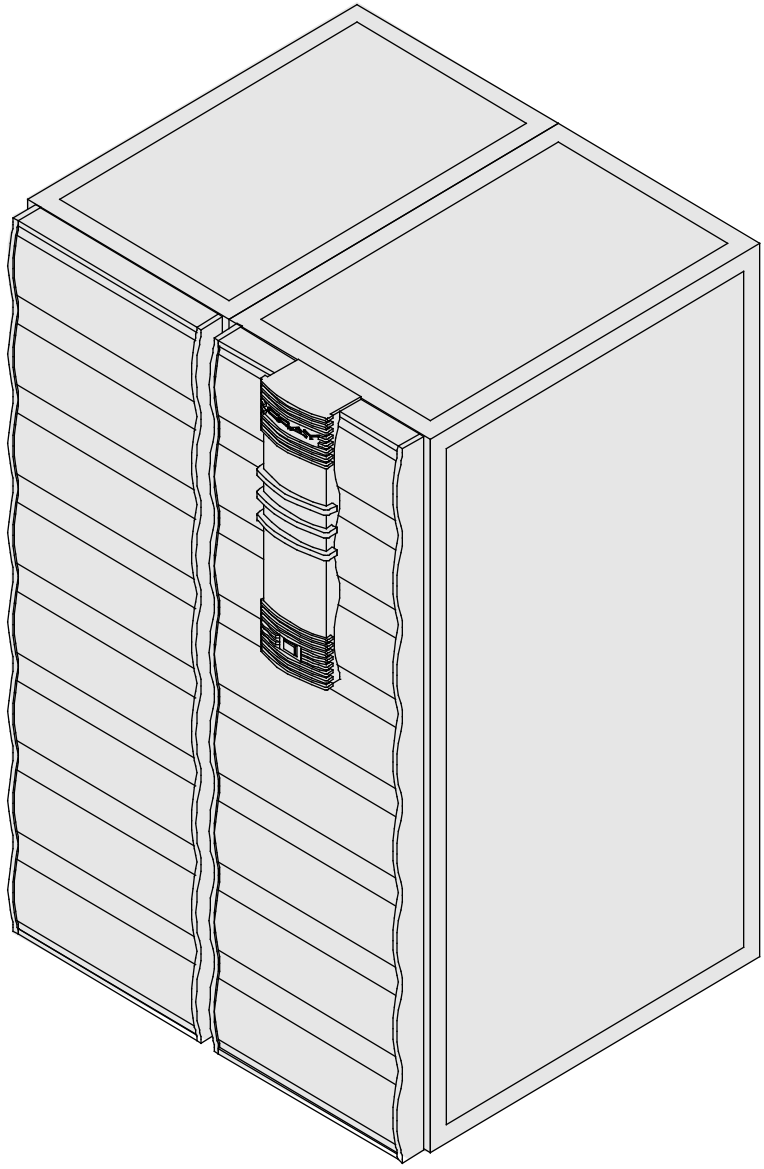
CRAY J916™ HI-P Upgrade Procedure



HMU-201-A

Cray Research Proprietary

Cray Research, Inc.

Record of Revision

REVISION	DESCRIPTION
	January 1995. Original printing.
A	December 1995. Table 2 revised.

Any shipment to a country outside of the United States requires a letter of assurance from Cray Research, Inc.

Requests for copies of Cray Research, Inc. publications should be directed to:

CRAY RESEARCH, INC.
Customer Service Logistics
1100 Lowater Road
Chippewa Falls, WI 54729

Comments about this publication should be directed to:

CRAY RESEARCH, INC.
Service Publications and Training
890 Industrial Blvd.
Chippewa Falls, WI 54729

CRAY J916 HI-P UPGRADE PROCEDURE

Getting Started

Device to Be Upgraded

This document contains procedures on how to add a High Performance Parallel Interface (HIPPI) pair (HI-P) to a CRAY J916 system.

Description of Upgrade

Cray Research customers have the option of adding an HI-P to their CRAY J916 system. This procedure is written to aid CRI support personnel in the task of installing HI-P upgrades. You may want to check off each step of this procedure as you complete it.

The HIPPI input channel (HI-I) and HIPPI output channel (HI-O) make up the HI-P. The HI-I may also be referred to as RA for receive or destination. The HI-O is also known as the TA for transmit or source.

The memory HIPPI implementation consists of a new device driver. The driver entry points are added to the character device switch table, major number 30. User programs access the memory HIPPI channels through character special device files that have a major device number of 30. This means that when creating device entries for the memory HIPPI you must use the major number of 30. HIPPI channels on IOS-E and IOS-D based systems are associated with major device number 7.

By convention, the device driver files are created in the `/dev/hippi0` directory. If a second HIPPI interface is present, `/dev/hippi1` would contain the special files for this interface. As is the case for other Cray Research HIPPI channels, two files are needed for full-duplex operation. A separate open system call is needed for an input and an output channel.

You can access the memory HIPPI in two ways:

- A user program can access the previously mentioned special files through the `open`, `close`, `read`, `reada`, `write`, `writea` and `listio` UNICOS system calls.

- You can configure the interface for use via TCP/IP.

The memory HIPPI driver implementation supports either dedicated use by one application (user program or TCP/IP) or shared usage. If the channel is being shared, all traffic must conform to the HIPPI-FP (framing protocol) specification

for packet format. When the input channel is run in shared mode, all sharing applications must try to keep multiple reads posted to the driver at all times. This ensures that a buffer is available for incoming data, because system buffering is not used for the HIPPI channel. The user program must post multiple reads when using either the `reada` or `listio` system call.

The driver automatically configures HIPPI channels during the UNICOS boot sequence. The memory HIPPI driver interrogates each potential memory HIPPI channel for the presence of this hardware by issuing the channel CPU master clear instruction followed by a read of the channel address. The hardware recognizes this sequence and causes it to return a channel identification that indicates the type of interface attached to the CPU in this position (a Y1 channel to the IOS, a memory HIPPI source, or a memory HIPPI destination).

The automatic channel configuration eliminates the need to make any manual UNICOS kernel configuration changes for memory HIPPI. To use the memory HIPPI, you must create the device nodes and update the TCP/IP configuration files; you do not have to reboot UNICOS for these activities.

When the memory HIPPI driver finds a channel during the boot sequence, a message is printed on the console. This message has the following format:

```
HIPPI input channel 054 initialized
HIPPI output channel 057 initialized
```

The preceding system console messages are also logged in files `/usr/adm/syslog/kern` and `/usr/adm/syslog/daylog`. The initialization code sets the memory HIPPI interrupt handler into the channel table. The numbers displayed depend on which CPU board the memory HIPPI resides. The following list depicts the channel numbers for each CPU.

```
CPU board                0   1   2   3
                        --  --  --  --
Output channel numbers:  27  37  47  57
Input channel numbers:   24  34  44  54

Output channel numbers:  23  33  43  53
Input channel numbers:   20* 30  40  50
```

NOTE: Channel 20 is the default deadstart channel for IOS 0.

HIPPI capable CPU modules can support up to four Y1 channels or four memory HIPPI channels, with the exception of CPU 0, which has the default deadstart channel for IOS 0. A 4-CPU CRAY J916 system can support up to seven memory HIPPI pairs (input and output), however only one HIPPI channel pair per CPU is recommended.

When the software detects a memory HIPPI module, it allocates table space to describe the channel and its 16 associated logical paths. The lowest-numbered channel is associated with minor device numbers 0 through 15 (major device number 30). The next higher channel is associated with minor device numbers 16 through 31. As is the case with the IOS-D and IOS-E, the dedicated path is the first logical path (path 0) for each channel. The remaining paths are shared paths.

Logical path 15 of each channel is reserved for only I/O control. This channel can be opened and closed and `ioctl` system calls can be issued, but data transfers will be rejected. Unlike the other shared channels, an open system call on path 15 does not prevent path 0 (the dedicated path) from being opened. This behavior is consistent with the way the HIPPI driver functions on Cray Research systems that have an IOS-E.

Specifying the `-ptp` option on the `ifconfig` command flags the interface as point-to-point. If you specify the `-ptp` option, the output HIPPI channel will be set into `HOLD` mode (not `DISCONNECT`). If you omit the `-ptp` option, the output HIPPI channel will be placed in `DISCONNECT` mode. You can set `HOLD` or `DISCONNECT` mode in this manner if the interface is associated with the dedicated path (path 0). You cannot change the disconnect flag from the system default if a shared path is chosen for TCP/IP. For memory HIPPI, `DISCONNECT` mode is the default. You set `HOLD` or `DISCONNECT` mode only on the output channel.

The memory HIPPI driver logs hardware errors into the file `/usr/adm/errfile`. You can specify the `-d hippi` option on the `/etc/errpt` command to process these entries.

The interface to the memory HIPPI driver is similar to the interfaces to the IOS-D and IOS-E HIPPI drivers. This similarity allows most programs that currently run on these platforms to use memory HIPPI without modification.

You may have to modify programs that take advantage of additional features. One of these features is the capability to read one packet into two user buffers or to write one packet from two buffers. On memory HIPPI, as on Cray Research systems that have an IOS-E, a maximum of two buffers is allowed (the IOS-D driver has no limit). The difference between the memory HIPPI and the IOS-E HIPPI is that on memory HIPPI, the first buffer must be 1024 bytes or smaller. Also, if an incoming packet has an initial short burst (allowed by the HIPPI standard specification) and this burst is smaller than the user's first buffer, the remainder of the first buffer will be skipped and the remainder of the packet will be written to the user's second buffer.

The `ioctl` system call is compatible with the IOS-E driver. The `ioctl` system call is used to change the values for time-outs, set a new I-field, change the disconnect flag, change the I-Field-in-data flag, or obtain a detailed status from a previous error.

Cray Research has developed drivers that support HIPPI disks and IPI-3 tape drives. This software will utilize the memory HIPPI driver to communicate to these HIPPI devices.

HI-P Upgrade Prerequisites

A processor module can support two HIPPI channel pairs. Channels 20/21 must be used as Y1 channels for proper CPU to IOS functions.

Selecting the HIPPI Ports

Your new HI-P should be installed on the highest-numbered processor module in your CRAY J916 system. You should choose the highest-numbered channel available. Refer to Table 1 and Figure 1 on page 7.

Special Considerations

If you have a processor module that is not HIPPI-compatible (P/Ns 90380000 and 90380001), you will receive a HIPPI-compatible processor module as part of your upgrade kit. This processor may or may not have the HIPPI paddle boards installed.

Table 1. CRAY J916 HIPPI Channel Configurations

Channel	Processor Number 0	Processor Number 1	Processor Number 2	Processor Number 3
	Paddle Card Slot J1			
Y1 Channel	*20/21	*30/31	40/41	50/51
HI-I Channel (RA)		*30/31	40/41	50/51
	Paddle Card Slot J2			
Y1 Channel	*22/23	*32/33	42/43	52/53
HI-O Channel (TA)		*32/33	42/43	52/53
	Paddle Card Slot J3			
Y1 Channel	24/25	34/35	44/45	54/55
HI-I Channel (RA)	24/25	34/35	44/45	54/55
	Paddle Card Slot J4			
Y1 Channel	26/27	36/37	46/47	56/57
HI-O Channel (TA)	26/27	36/37	46/47	56/57

* Either paddle card 0 or paddle card 1 can be configured as the deadstart channel. By default, paddle card 0 20/21, Y1, is configured as the deadstart channel.

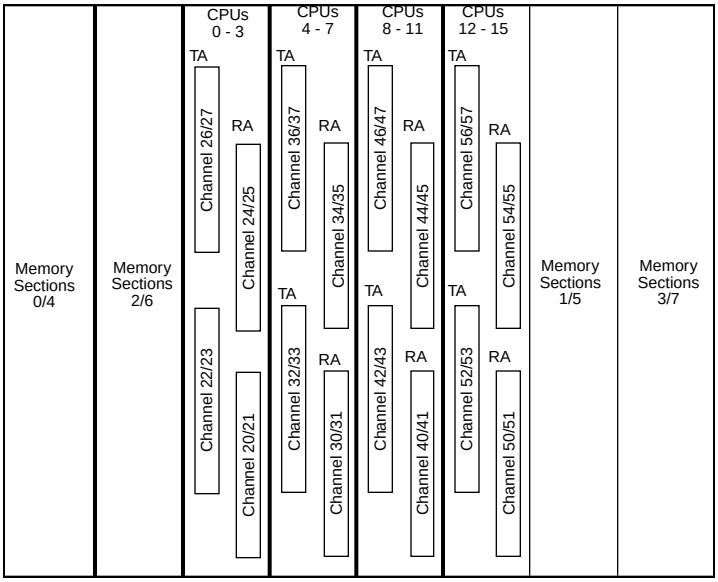


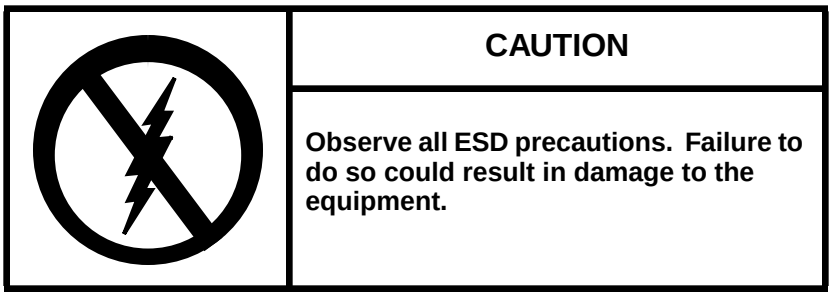
Figure 1. CRAY J916 Processor Module Channel Assignments

Training Requirements

Cray Research personnel who perform this upgrade must have completed training in CRAY J916 hardware and system administration. If this is not possible, a hardware-trained person must have a system administrator available during this upgrade. Experience in upgrading or installing the UNICOS operating system on a CRAY J916 system or CRAY Y-MP EL system is required. Please read this document thoroughly before performing the upgrade. You should understand the concepts and terminology related to HIPPI by reviewing the *HIPPI Channel Administrator's Guide*, publication number SG-2159.

ESD Precautions

Observe ESD precautions during the entire upgrade process. Required apparel includes an ESD smock and an ESD wrist strap. Do not wear watches or jewelry when you work on a CRAY J916 system cabinet.



ESD Smock

Wear a Cray Research-approved static-dissipative smock when servicing or handling an ESD-sensitive device. Completely button the smock and wear it as the outermost layer of clothing. You must have a portion of the smock's sleeves in direct contact with the skin of your arms. Skin contact is essential for a dissipative path-to-earth ground through your wrist strap. Tuck hair that exceeds shoulder length inside the back of the smock.

Wrist Strap

Wear a Cray Research-approved wrist strap when handling an ESD-sensitive device to eliminate possible ESD damage to equipment. Connect the wrist strap cord directly to earth ground.

Reference Publications

Refer to the following publications if you have questions when performing this upgrade:

- *UNICOS Basic Administration Guide for CRAY J90 and CRAY EL Series*, Cray Research publication number SG-2416
- *Automated Confidence Testing,* Cray Research publication number HDM-110-0
- *CRAY J916 Service Manual Kit,* Cray Research publication number HMK-101-0
- *UNICOS Installation and Configuration Tool Reference Manual*, publication number SR-3090
- *HIPPI Channel Administrator's Guide*, publication number SG-2159
- *UNICOS System Administration*, publication number SG-2113

Estimated Time to Install Upgrade

Table 2 divides the HI-P upgrade process into four separate procedures. Use this table to determine how much system time you should request to complete this upgrade. Table 3 lists the parts required for the upgrade.

NOTE: It is recommended that you contact the customer and request that he or she complete a full backup prior to this upgrade.

Table 2. Estimated Times to Install Upgrade

Install Task	Estimated Time to Install Upgrade
Hardware Install	1.5 hours
Hardware Verification Testing	1 hour
Software Configuration	3 hours
HIPPI Verification Testing	1 hour

Parts Required

Table 3. HI-P Kit Contents

CRI Part Number	Description
90288000	Module Assy, HIPPI Destination
90287500	Module Assy, HIPPI Source
90390900	Cable Assy, HIPPI-CPU, TA
90390901	Cable Assy, HIPPI-CPU, RAC
90216400	Cable Assy, HIPPI, Ext Cab (2 cables)

Tools Required

Most of the tools required for performing upgrades are common hand-held tools that are included with the Customer Service toolkit.

Software Required

- Minimum IOS kernel revision - N/A
- Minimum UNICOS revision - 8.0.4

Conventions

The following conventions are used throughout this document:

Convention	Meaning
`command`	This fixed-space font denotes literal items such as commands, files, routines, path names, signals, messages, and programming language structures.
`manpage`(*x*)	Man page section identifiers appear in parentheses after man page names.
variable	Italic typeface denotes variable entries, words or concepts being defined.
`user input`	This bold fixed-space font denotes literal items that the user enters in interactive sessions. Output is shown in nonbold, fixed-space font.
<KEY>	This convention indicates a key on the keyboard.

Getting Started

It is recommended that you check off each step as it is performed. It also is suggested that you inventory the contents of the upgrade kit before you start this procedure.

Create a Backup Copy of the UNICOS File System

It is recommended that you create a backup copy of the UNICOS file system before you proceed with the upgrade procedures. Refer to the *UNICOS Basic Administration Guide for CRAY J90 and CRAY EL Series*, publication number SG-2416, for details on how to create a backup copy of the UNICOS file system.

Power Down the CRAY J916 System

1. Ensure that the customer has brought the system to single-user mode.
2. Using the right mouse button, click on any open working space. The `Workspace` menu will appear.
3. From the `Workspace Menu`, select the `J90 Console` menu item.

NOTE: You must have super user privileges to perform Step 4.

4. Log into the UNICOS operating system by entering `<CONTROL-a>` to get a UNICOS prompt and enter the root login and password.
5. Shut down the UNICOS operating system by entering the following commands at a UNICOS prompt:

```
# cd /
# /etc/shutdown 120        (executes after 120 seconds)
# /bin/sync
# /bin/sync
# /bin/sync
# /etc/ldsync              (if you are using ldcache)
```

6. Stop the `J90 Console` connection by entering the following commands:

```
# <CONTROL-a>              (toggles to the IOS)
sn9xxx-ios0> mc
sn9xxx-ios0> reset         (takes 30-45 seconds
to execute)
BOOT[sn9xxx-ios0]> ~. <CONTROL-c>
```

7. Open the mainframe cabinet front door by pushing down on the latch and swinging the door open.

8. Power off the system by pressing the CCU SYSTEM OFF button.

Hardware Upgrade Procedure

Open Mainframe Cabinet Rear Door

1. At the rear of the mainframe cabinet, turn the two door fasteners 1/4-turn counterclockwise with a 5/32-in. allen wrench.

2. Grasp the door handle and swing the door open to the right.

Open the I/O Cabinet Rear Door

1. At the rear of the I/O cabinet, turn the two door fasteners 1/4-turn counterclockwise with a 5/32-in. allen wrench.

2. Grasp the door handle and swing the door open to the right.

If you received a processor module with your upgrade kit, you may proceed to "Prepare Processor Module for Paddle Board Installation" or if the HIPPI paddle cards were shipped with the HIPPI paddle boards installed, proceed to "Install HI-P cables."

Remove Processor Module That the HI-P will be Installed on

1. Remove the processor module from the card cage.

    a. Turn the jack screws located at the top and bottom of the module counterclockwise until the module is loose in the chassis.

b. Grasp the module securely and remove it from the chassis. The modules are heavy. Use proper body mechanics to avoid back injury and module damage.

CAUTION
The modules are heavy. Use proper body mechanics to avoid back injury and module damage.

2. Lay the processor module on its side on a static-protected work surface.

Prepare Processor Module for Paddle Board Installation

Use the following procedure for both the TA (HI-O) and the RA (HI-I) paddle board. Install the HI-O first; then simply flip the module and repeat the following steps to install the HI-I.

1. Orient the module as shown in Figure 2.
2. Remove the 15 screws (11 on the side and 4 on the front) from the processor module that secure the metal side cover in place.
3. Remove the four screws that secure the cover to the front of the processor module. Refer to Figure 2.
4. Remove the cover from the processor module by lifting straight up. Set the cover aside.

CAUTION
Ensure that no debris is allowed to fall into the processor module. Such debris could cause module damage by creating short circuits.

5. Remove the five securing screws (with lockwashers) that secure the connector faceplate cover. Retain these screws.

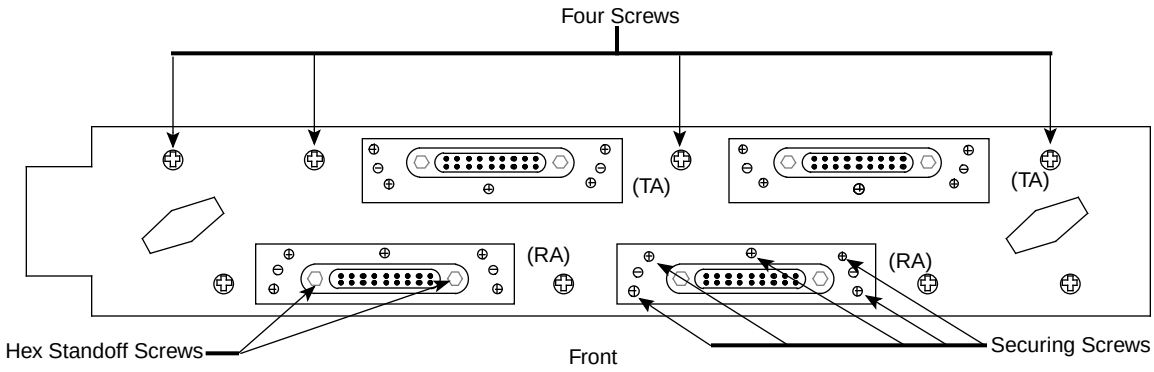


Figure 2. Connector Hardware Screws

Install the New HI-O or HI-I Paddle Board

1. Unpack the HI-I (RA) (P/N 90288000) or HI-O (TA) (P/N 90287500) paddle board and inspect it for damage. Save the packing material for reuse. Install the HI-O first; then flip the module and install the HI-I.

2. Remove the processor module's blank faceplate connector cover.

   a. Remove the two cheese-head screws from the rear (noncomponent) side of the paddle board. Refer to Figure 3.

   NOTE: The cheese-head screws are located on the noncomponent side, and the bracket that they screw into is located on the component side of the board.

   b. Remove the two hex standoff screws from the front of the connector faceplate. Refer to Figure 2. (These screws secure the cable to the paddle board.)

   c. Remove the two plastic standoff screws from the processor module and retain them for reuse.

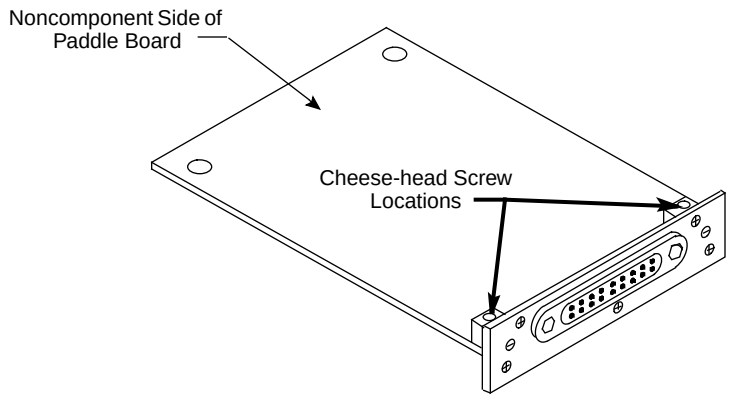


Figure 3. Paddle Board Cheese-head Screw Locations

3. Align the 128 pins on the paddle board with the 128 sockets on the processor module and press the paddle board into position.

4. Replace the connector faceplate and replace and tighten the two cheese-head screws at the rear (noncomponent side) of the paddle board.

5. Replace and tighten the two hex standoff screws on the front of the connector faceplate.

6. Replace the two plastic standoff screws that you removed from the processor module in Step 2, Step c.

7. Replace the side cover of the processor module.

    a. Insert and tighten the 15 screws to secure the cover to the side of the processor module.

    b. Insert and tighten the four screws to secure the cover to the front of the processor module.

8. Repeat Step 1 through Step 7 for the other paddle board.

9. Insert the processor module into the card cage.

    a. Place the module into the module guides in the mainframe chassis and push the module into the chassis until it contacts the air damper control handle.

b. Open the slot air damper to its open position by turning the air damper handle one-fourth turn counterclockwise. Continue to push the processor module into the chassis until it contacts the backplane.

c. Tighten the two jack screws simultaneously until the module is fully seated.

Install HI-P Cables

1. Remove two I/O bulkhead plates from any slot labeled A1 through A11.
2. Insert the end of the HI-I cable (P/N 90390901) that has the I/O bulkhead faceplate attached into the rightmost empty I/O bulkhead position.
3. Insert the HI-O cable (P/N 90390900) into the other empty I/O bulkhead position. [The HI-O (TA) cable should always be installed to the left of the HI-I (RA) cable.]
4. Extend each of the cables up the right side of the I/O cabinet just to the right of the power distribution rail.
5. Route these cables through the I/O cabinet to the mainframe cabinet.
6. Install, finger tight, each HIPPI cable into its labeled position in the processor module connector ports.
7. Take up any slack in each of these cables and curl the excess cable above the CPU card cage and secure it with tie wraps.

NOTE: Depending on the number of cables already installed for other devices, the HI-P cables can be routed up either side of the I/O cabinet.

Power Up the System

NOTE: If the `J90 Console` window is already up, skip to Step 3.

1. Using the right mouse button, click on any open working space. The `Workspace` menu will appear.
2. Select the `J90 Console` menu item.
3. Move the circuit breaker on the back of the I/O cabinet to the `ON` position first, and then move the circuit breaker on the mainframe cabinet to the `ON` position.

4. Press the `Alarm Acknowledge` button on the CCU.
5. Press the `CPU RESET` button on the CCU.
6. Press the `VME RESET` button on the CCU.
7. Verify that the SYSTEM READY light on the control panel illuminates.
8. Close the mainframe front door.

Verify That the Upgraded Hardware is Functional

The second level of ACT, the menu system, provides a menu-driven interface that selects and runs specific diagnostics. If ACT detects a failure, refer to *Automated Confidence Testing*, Cray Research publication number HDM-110-0.

1. At the `sn9`*xxx*`-ios0>` prompt, invoke the ACT menu system by entering the following command:

   ```
   act_menu
   ```

2. Select 1 to run all basic tests once the `Automated Confidence (BASIC) Test Menu` appears.
3. Select `n` from the `Automated Confidence (BASIC) Test Menu` to go to the `Automated Confidence (INTERMEDIATE) Test Menu.`
4. Select `1` to run all intermediate tests.
5. Select `n` from the `Automated Confidence (INTERMEDIATE) Test Menu` to go to the `Automated Confidence (COMPREHENSIVE) Test Menu.`
6. Select `1` to run all comprehensive tests.
7. Select `q` to quit the ACT menu system.

Software Change Procedure

Installation Procedure

Perform the following procedure to use the UNICOS Installation / Configuration Menu System (ICMS) to rebuild the UNICOS operating system. You must have super user privileges. For additional information on the ICMS, see the *UNICOS Installation and Configuration Tool Reference Manual*, publication SR-3090.

NOTE: If you have not already done so, it is recommended that you create a backup copy of the UNICOS file system.

1. Save the existing `/sys/param` file by entering the following commands:

```
sn9xxx-ios0> cd /sys
sn9xxx-ios0> cp param param.old
```

2. Start the UNICOS operating system by entering the following command:

```
sn9xxx-ios0> boot
```

NOTE: The following message will be displayed if the memory HIPPI channels are recognized (the exact channel numbers will depend on the CPU and Y1 ports on which the memory HIPPI paddle boards are installed). If the following messages are **not** displayed, DO NOT PROCEED. Verify that the correct number of CPUs are configured in the `param` file and that the CPU board is seated properly.

```
HIPPI input channel 034 initialized
HIPPI output channel 037 initialized
```

3. Enter multiuser mode by entering the following command (for more information on bringing your system to multiuser mode, see the *UNICOS Basic Administration Guide for CRAY J90 and CRAY EL Series*, publication SG-2416):

```
# /etc/init 2
```

4. Log on as super user (root).

5. To ensure that the `/etc/config/param` file is up-to-date, copy it from the IOS disk to the UNICOS file system by entering the following command:

```
# exdf -i /sys/param > /etc/config/param
```

6. Enter the UNICOS Installation / Configuration Menu System by entering the following command:

```
# /etc/install/install
```

7. It is recommended that before you configure your system with the UNICOS ICMS, you should import the current system configuration into the ICMS. Select the following menu:

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Utilities
.   .   Import Utility
.   .   .   Import Options
```

8. Set the Import Options to the following values:

```
   Import Options
S->Import root mount point
   Stop import on error?          YES
   Import host or guest versions? host
   Reload default import table ...<
```

9. Select the `Import Table` menu and set `Import?` to `YES` for the following Import Table entries (this table may be multiple pages):

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Utilities
.   .   Import Utility
.   .   .   Import Table
            Import Table
```

Class	Description	Import?	Program	Options
E->HARDWARE	Param	**YES**	hdwparam.sh	-i $RELEA
KERNEL	Config.h uts	**YES**	utsconfh.sh	-i $RELEA
KERNEL	Param uts	**YES**	utsparam.sh	-i $RELEA
KERNEL	Comm channels	**YES**	utscparam.sh	-i $RELEA
HOSTS	Hosts	**YES**	utlimp.sh	hosts
NETWORKS	Networks	**YES**	utlimp.sh	networks
NETIF	Network Interfaces	**YES**	netifs.sh	-i
NETHWADDR	Network H/W Address	**YES**	utlimp.sh	hycf

10. Select the `Import Utility` menu and type `ALL` in the `Import Class to run` entry:

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Utilities
.   .   Import Utility
                  Import Utility

    Import options ==>
    Import table ==>
S-> Import class to run     ALL
    Run the import process...
```

11. Select `Run the import process` and answer yes `(y)` to the question to overwrite all or parts of the menu system database. Answer `y` to the following question:

```
Do you want to continue? (y/n)   y
```

12. Select the `Major Software Configuration` menu. Verify that the `HSX/high-speed/HIPPI device support` is `on`. If it is labeled `off`, then change the value to be `on`.

```
 UNICOS 8.0 Installation / Configuration Menu System
 .   Configure System
 .   .   Major Software Configuration

         Major Software Configuration

 S-> HSX/high-speed/HIPPI device support     on
```

13. Select the `High-speed Channel Configuration` menu. Enter the following parameters to build the HIPPI device node entries. The following is a sample entry:

```
   UNICOS 8.0 Installation / Configuration Menu
   System
   .   Configure System
   .   .   UNICOS Kernel Configuration
   .   .   .   Communication Channel Configuration
   .   .   .       High-speed Channel Configuration

                   High-speed Channel Configuration

              S->High-speed device (hidev) ordinal0
                 I/O cluster (IOC) number
                 IOP number
                 Channel number
                 Device Flags
                 I/O direction                output
                 Interface type
```

Definitions:

`High-speed device (Hidev) ordinal` describes the ordinal of the High-speed network device. Each High-speed device is addressed as "device number *n*," where "*n*" is an ordinal from zero (0) to 15. This number must be unique for each I/O direction defined for the high-speed device (for example, 0 for input, 1 for output, 2 for input, 3 for output, and so on).

`I/O cluster` (IOC) number is not used for the memory HIPPI.

`IOP number` is not used for the memory HIPPI.

`Channel number` is not used for the memory HIPPI.

`Device Flags` is not used for the memory HIPPI.

`I/O direction` indicates the channel's direction of input or output.

`Interface type` is not used for the memory HIPPI.

The memory HIPPI channel requires an input and output device to be configured to send and receive data. Below is an example configuration for two HIPPI channel pairs (input and output for each HIPPI channel).

```
                High-speed Channel Configuration

          HiDev  Ioc  Iop  Chn  Flgs  I/O     Type
          -----  ---  ---  ---  ---- -----   -------
          0                          input
          1                          output
          2                          input
      E-> 3                          output
```

14. Exit the `High-speed Channel Configuration` menu and answer `y` to the following question:

```
Do you want to update form file? (y/n):    y
```

15. The High-speed device directory and file modes are defaulted to allow direct access only from a root process. To change these values, select the `Communication Channel Configuration` menu. Change the following entries:

    NOTE: Typical values that are used to allow user processes to gain access to the `/dev/hippi` device files are 0755 for the High-speed dev. directory mode (`hidirmode`) and 0666 for High-speed dev. file mode (`hifilemode`).

```
 S-> High-speed dev. directory mode (hidirmode) 0700
     High-speed dev. file mode (hifilemode)     0600
```

16. Select the `UNICOS Kernel Configuration` menu. Activate the kernel configuration by selecting the `Activate the kernel configuration` option.

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   UNICOS Kernel Configuration
```

Answer `y` to the following question:

```
Do you want to proceed with the configuration
update? (y/n)  y
```

17. Step 18 through Step 27 describe how to configure the memory HIPPI interface for TCP/IP. If the memory HIPPI will not run TCP/IP on your system, continue with Step 28.

18. Select the `Network Parameters` menu and enter the following parameters to configure the memory HIPPI interface(s):

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   UNICOS Kernel Configuration
.   .   .   Network Parameters
```

NOTE: The following parameters represent a portion of the Network Parameters display but are the only parameters that must be updated to install the memory HIPPI interface(s).

```
                    Network Parameters

 S-> Number of TCP memory buffers (TCP_NMBSPACE)4000
     Max. high-speed network devices (himaxdevs)4
     Max. high-speed paths in system (himaxpaths)8
```

Definitions:

`Number of TCP memory buffers` refers to the number of mbufs to be defined for the system. This value is system-dependent. If your system already runs with 4000 mbufs or your system is configured for NFS and/or gated, you may have to monitor mbuf usage and increase the amount if needed. To determine the correct number of mbufs for your system, read subsection 23.3.2,

Buffering and Memory Requirements, in *UNICOS System Administration*, publication number SG-2113. Each time you change this value, you **must** reboot UNICOS, because the kernel allocates memory for mbufs at boot time.

`Maximum high-speed network devices (himaxdevs)` sets the maximum high-speed network devices that can be open in the system at one time.

`Maximum high-speed paths in system(himaxpaths)` sets the maximum logical high-speed paths that can be open in the system at one time.

19. Select the `UNICOS Kernel Configuration` menu. Activate the kernel configuration by choosing the `Activate the kernel configuration` option.

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   UNICOS Kernel Configuration
```

    Answer **y** to the question `Do you want to proceed with the configuration update? (y/n)`.

20. Select the Host Address Configuration menu. Configure the memory HIPPI internet connection(s) for TCP/IP by creating entries for each host that will be connected on the HIPPI network.

    NOTE: The host file contains all network adaptors and destination devices.

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   Network Configuration
.   .   .   General Network Configuration
.   .   .   .   Host Address Configuration
```

```
      Host Address Configuration

Proto     Name          Address     Comment
-----     ----------    ----------  --------->
inet      yourhost-hippi    128.162.102.25
inet      remotehost-hippi  128.162.102.27
```

Definitions:

`Proto` should be `inet`.

`Name` is the IP host address alias.

`Address` is the standard IP address value.

21. Select the `Network Address Configuration` menu. Configure the memory HIPPI internet network address for TCP/IP by creating an entry for each HIPPI network. There will be a network address for each HIPPI channel pair that is being installed that will communicate via TCP/IP.

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   Network Configuration
.   .   .   General Network Configuration
.   .   .   .   Network Address Configuration

Network Address Configuration

   Proto     Name          Address      Comment
   -----     ----------    ----------   --------->
E->inet      yourhippinet  128.162.94
```

Definitions:

`Proto` should be `inet`.

`Name` is the IP host address alias.

`Address` is the standard IP address value.

22. Exit the `Network Address Configuration` menu and answer `y` to the following question:

```
Do you want to update form file? (y/n):   y
```

23. Select the `Network Hardware Address Configuration` menu. Configure the memory HIPPI `hycf` file by creating a new entry for each node on the HIPPI network (see the Definitions that following for correct entry information).

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   Network Configuration
.   .   .   General Network Configuration
.   .   .   .   Network Hardware Configuration

    Network Hardware Address Configuration

  HYCF group suffix                    hippi
S->Direct or gateway?                  direct
  Host name or address                 yourhost-hippi
  Destination hardware address         01000000
  Control value (HIPPI input minor #)  0000
  Access code (HIPPI output minor #)   0010
  Maximum Transmission Unit (MTU) size 65536
  Gateway 1
  Gateway 2
  Gateway 3
  Gateway 4
  Gateway 5
  Gateway 6
  Gateway 7
  Gateway 8
```

Definitions:

`HYCF` group suffix is a suffix specifying the local network to which the host specified in this submenu is attached. (All entries with a common group will be placed in the same `hycf.*` file, with the group as the suffix.) (NOTE: If the networking interfaces are being maintained via the Network Interface Configuration menu, the value in this selection should correspond to the value in the second field of that menu. In other words, this field specifies which `/etc/hycf.*` in which the entry will be placed, and the second field of the Network Interface Configuration menu specifies which `/etc/hycf.*` will be loaded via `hyroute`(8), to initialize hardware addresses for an interface.)

`Direct or gateway` should be set to `direct`.

`Host name or address` should contain the host name or Internet address of the host for which the hardware address is being configured.

`Destination hardware address` should be the actual hardware address of the specified host and should be specified in hexadecimal. This is referred to as the `I-field` for the HIPPI switch, and should be set as follows:

```
CCRRRRRR
```

`CC` = Control Information
`RRRRRR` = Routing information

`Destination hardware address` should be set to 0 for a directly connected HIPPI. A directly connected HIPPI connects two HIPPI nodes directly without going through a HIPPI switch. Refer to the *HIPPI Channel Administrator's Guide*, publication number SG-2159, for more information regarding the HIPPI I-field. If you are connecting to a HIPPI switch, you must refer to the manufacturer's HIPPI switch documentation for I-field decoding. The memory HIPPI connection does not support 64-bit mode. If this bit is set in the I-field, the connection will fail.

`Control value (HIPPI input minor #)` and `Access code (HIPPI output minor #)` in the `/etc/hycf.hippi` file reflect an index into the `comm` table created by the memory HIPPI driver software and point to the minor numbers for the HIPPI device files. These fields are in hexadecimal format and significant only for the resident HIPPI connection entry. See the `/etc/hycf.hippi` file example on page 31 for a description on how these values correspond to the HIPPI device files.

`Maximum Transmission Unit (MTU) size` should be set to 65536 for the memory HIPPI interface.

24. Exit the `Network Address Configuration` menu and answer `y` to the following question:

```
Do you want to update form file? (y/n):   y
```

25. Select the `Network interface configuration` menu. Configure the memory HIPPI in the `/etc/config/interface` file (see the Definitions that following for correct entry information).

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   Network Configuration
.   .   .   Network Interface Configuration

S->Interface name                        hi0
  HYCF file (path name or hycf. suffix)hippi
  Address family                         inet
  Interface address                      yourhost-hippi
  Destination address (point-to-point)
  Subnetwork mask                        0xffffff00
  Interface type                         hippi
  Broadcast address
  MTU size mask                          65536
  Read buffer size
  Write buffer size
  Use packet trailers?                   NO
  Address Resolution Protocol available?NO
  Debug mode?                            NO
  Retry in background?                   NO
  Is this an alias?                      NO
  Routing metric
  Minimum security level
  Maximum security level
  Minimum security compartments
  Maximum security compartments
  Authority bytes
  NSAP NSEL length in bytes (OSI)
  Channel access requires privilege
```

Definitions:

`Interface name` should specify the interface name and number. The first HIPPI channel pair will be `hi0`, the second will be `hi1`, and so on.

`HYCF file (path name or hycf. suffix)` is the suffix name of the `hycf` file. In the preceding example, the `hycf` file is `/etc/hycf.hippi` (`/etc` is the default directory).

`Address family` should be `inet`.

`Interface address` is the local Internet address or host name of the interface.

`Destination address (point-to-point)` should not be set for connections to a HIPPI switch. For a directly connected HIPPI between two HIPPI nodes, this should be the Internet address of the destination host. This will default the HIPPI connection to hold mode.

`Subnetwork mask` should be the mask used to separate the subnetwork number from the host portion of the address if the network this interface is connected to is a subnetted network.

`Interface type` should be set to `hippi`.

`Broadcast address` is not used.

`MTU size mask` is the size of the read buffers posted to the low level driver. This value should be set to 65536.

`Read buffer size` is the maximum number of buffers that can be posted to the HIPPI driver for reads. The default value is three read buffers. TCP/IP will post three reads when the interface is configured up.

`Write buffer size` is the maximum number of buffers that can be posted to the driver for writes. The default value is 36 write buffers.

`Use packet trailers` should be set to `NO`.

`Address Resolution Protocol available` should be set to `NO`.

`Debug mode` should be set to `NO.`

`Retry in background` should be set to `NO`. This is set to `YES` if you want retry attempts to initialize this interface to take place in the background.

`Is this an alias` should be set to `NO`.

`Routing metric` is optional. If used, it should specify the routing metric for the interface, where a higher metric indicates a less favorable route.

26. Exit the Network Interface Configuration menu and answer `y` to the following question:

```
Do you want to update form file? (y/n):   y
```

27. Select the `General Network Configuration` menu. Activate the general network configuration by selecting the `Activate general network configuration` option.

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Configure System
.   .   Network Configuration
.   .   .   General Network Configuration
```

Answer `y` to the following question:

```
Do you want to proceed with the configuration
update? (y/n)  y
```

28. Copy the param file to the console disk. Select the following menu:

```
UNICOS 8.0 Installation / Configuration Menu
System
.   Utilities
.   .   Expander File Transfers
```

29. Verify that the following parameters are configured to transfer the `/etc/config/param` file to the console disk:

```
     Expander File Transfers

S->Transfer UNICOS kernel to the expander?NO
Transfer CSL param file to the expander?YES

Expander directory name              sys
Expander file name suffix

Do the transfer to the expander ...
```

30. Execute the transfer by selecting the following option:

```
A-> Do the transfer to the expander ...
```

31. Exit the Installation / Configuration Menu System by typing **`q`** and answering yes (**`y`**) to the question `Do you want to quit`?.

Configuration File Examples

The UNICOS Installation / Configuration Menu System (ICMS) updates the files described in this section. Exact parameter settings will be system-dependent. If TCP/IP was not configured, only the `/etc/config/param` file will be updated by the ICMS.

NOTE: Only the part of the files that refer to the memory HIPPI interface and what was configured with the ICMS are shown in the file samples in this section.

1. The `/etc/config/param` file details the parameters needed to configure the memory HIPPI interface(s) for the host system. It contains the following information:

```
network {
   4000 tcp_nmbspace;
        4 himaxdevs;
        8 himaxpaths;

        0700 hidirmode;
        0600 hifilemode;
```

2. The `/etc/config/interfaces` file contains the parameters to configure up the memory HIPPI interface(s) with the `/etc/initif` script. It contains the following information:

```
# File format is:
#
# name  hycf_file   family  address    pt-to-pt-dest   args:
#                                      netmask
#                                      iftype
#                                      broadcast
#                                      mtu
#                                      rbuf
#                                      wbuf
#                                      bg
#                                      hwloop
#
lo0     -      inet yourhost-
en0     -      inet yourhost-ether-          netmask 0xffffff00
fddi0   -      inet yourhost-fiddi-          netmask 0xffffff00
hi0     /etc/hycf.hippiinet yourhost-hippi -netmask 0xffffff00
iftype hippi mtu 65536
```

3. The `/etc/hosts` file contains the following information:

```
128.162.95.23     yourhost-hippi
128.162.95.24     remotehost1-hippi
128.162.95.25     remotehost2-hippi
128.162.95.26     remotehost3-hippi
```

4. The `/etc/hycf.hippi` file contains the following information:

```
        Hostname           I-field    In     Out    MTU
        --------           -------    ---    ---    ---
direct yourhost-hippi      01000000   0000   0010   65536;
direct remotehost1-hippi   01000000   0000   0010   65536;
direct remotehost2-hippi   01000000   0000   0010   65536;
direct remotehost3-hippi   01000000   0000   0010   65536;
```

The `In` and `Out` numbers in the `/etc/hycf.hippi` file reflect an index to the `comm` table created by the memory HIPPI driver software and point to the minor numbers for the HIPPI device files. This field is in hexadecimal format and is significant **only** for the resident HIPPI connection entry. For example, the first entry (the resident CRAY J90 series connection) in the preceding entry has `0000` for the `In` and `0010` for the `Out` minor numbers.

The `In` minor number of `0000` corresponds to the input of the HIPPI device files (in the following example it is `/dev/hippi0/i00`):

```
# ls -l /dev/hippi0/i00
crw-rw-rw- 1 root root 30, 0 May 7 10:49 /dev/hippi0/i00
```

The `Out` minor number of `0010` (0010 hexadecimal or 16 decimal) corresponds to the output HIPPI device file (in the following example, it is `/dev/hippi0/o00`):

```
# ls -l /dev/hippi0/o00
crw-rw-rw- 1 root root 30, 16 May 28 10:34 /dev/hippi0/o00
```

5. The /etc/networks file contains the following information:

```
yourhippinet      128.162.95
```

6. The `/dev/hippix` directory and device files (*x* = HIPPI Interface number) contain the following information:

NOTE: If you have multiple HIPPI interfaces, you must specify the device entry directories separately. The following sample shows two HIPPI channels; the device files for interface #0 will be in `/dev/hippi0`

and the device files for interfaces #1 will be in `/dev/hippi1`.

```
HIPPI     Device       Major
Interface Directory    Number   Device Files    minor numbers
--------- ----------   ------   -------------- -------------
#0        /dev/hippi0  30       i00 thru i15   0 thru 15
          /dev/hippi0  30       o00 thru o15   16 thru 31

#1        /dev/hippi1  30       i00 thru i15   32 thru 47
          /dev/hippi1  30       o00 thru o15   48 thru 63
```

Shut Down UNICOS and Reboot the System

Shut down UNICOS and reboot the system by executing the following commands:

1. Shut down the UNICOS operating system by entering the following commands at a UNICOS prompt:

```
# cd /
# /etc/shutdown 120      (executes after 120 seconds)
# /bin/sync
# /bin/sync
# /bin/sync
# /etc/ldsync                (if you are using ldcache)
# <CONTROL-A>                (toggles to the IOS)
sn9xxx-ios0>
```

2. Reload the IOS by entering the following command:

```
sn9xxx-ios0> reload
```

3. Reboot the system to single-user mode by entering the following command:

```
sn9xxx-ios0> boot
```

NOTE: The following message will be displayed if the memory HIPPI channels are recognized (the exact channel numbers will depend upon the CPU and Y1 ports on which the memory HIPPI paddle boards are installed):

```
HIPPI input channel 034 initialized
HIPPI output channel 037 initialized
```

4. Enter multiuser mode in UNICOS by entering the following command:

```
# /etc/init 2
```

NOTE: Questions will be displayed and will require input before UNICOS will be completely booted to multiuser mode. Refer to the *UNICOS Basic Administration Guide for CRAY J90 and CRAY EL Series*, publication number SG-2416, for specific information regarding these messages.

HIPPI Verification Testing

Test the Raw HIPPI Interface in Loopback

To test the raw HIPPI interface, complete the following steps:

1. Configure TCP/IP down for the memory HIPPI interface (*x*) by entering the following command:

```
# /etc/ifconfig hix down
```

2. Connect one end of a HIPPI cable to the input port of the memory HIPPI board by connecting the cable to the DESTINATION connector (HI-I RA) at the bulkhead connector.

3. Connect the other end of the HIPPI cable to the output port of the memory HIPPI board by connecting the cable to the SOURCE connector (HI-O TA) at the bulkhead connector.

4. With the cable connected in loopback (from the SOURCE connector to the DESTINATION connector of the memory HIPPI board), test the raw HIPPI connection by using VHT. An example follows. (NOTE: The last parameter in the following command is the number one):

```
# /etc/vht -i /dev/hippix/i00 -o /dev/hippi0/o00 -c1000 -D -P -1
```

This example uses the input device `/dev/hippi0/i00` and the output device `/dev/hippi0/o00`. The `-c1000` option limits the pass count to 1000 passes. The `-D` option specifies to disconnect between packets. The `-P -1` option specifies a pseudorandom data test pattern. The following example shows a successful VHT execution:

```
# /etc/vht -i /dev/hippi0/i00 -o /dev/hippi0/o00 -c1000 -D -P -1
 Using I-field 00000000
 random pattern selection
 HIPPI test path 0 completed pass XXX

/etc/vht 0 passes 1-1000 no fatal errors
#
```

NOTE: The *XXX* is the pass count and will increment as VHT executes.

Connect HI-P to Customer's External Device

1. Connect the HI-O (*Transmit - TA - Source*) cable from the CRAY J916 I/O bulkhead to the input (*Receive - Destination*) of the external device.

2. Connect the HI-I (*Receive - RA - Destination*) cable to the output (*Transmit - Source*) of the external device.

Node-to-Node Raw HIPPI Test

To complete the node-to-node raw HIPPI test, complete the following steps:

1. If you connect the memory HIPPI to a HIPPI switch, you can perform loopback to the switch by including an I-field. The I-field value depends on the switch configuration.

   An example of the VHT command line with the I-field specified follows:

```
# /etc/vht -i /dev/hippi0/i00 -o /dev/hippi0/o00 -c1000 -D -P -1 -I 0x01000004
```

2. If the CRAY J916 memory HIPPI is connected to another Cray Research HIPPI interface, you can perform a master/slave test by using VHT.

NOTE: The location and name of the device files on another Cray Research system may be different from those described here.

To run this test, you either must be logged on to both Cray Research systems and have access at one central point (that is, using multiple screens or windows) or you must be able to run a remote shell from the Cray Research system that initiates the VHT test to the other Cray Research system over another network media (for example, Ethernet or FDDI). This requires you to set up a `.rhosts` file in the `$HOME` directory of the remote host.

The following example uses VHT to execute a remote shell command to start the master/slave VHT test. To execute this, you must have the necessary `/etc/hosts` entries for the remote Cray Research system host specified by the `-h` option or you can specify the IP dot notation address. If you use an alias for the remote host address, you may want to reference Step 20 in the Installation Procedure for instructions on updating the `/etc/hosts` file.

NOTE: For the following example, the VHT command is executed on CrayA. `CrayB-ether` is an Ethernet connection to another Cray Research computer, CrayB, that is directly connected on the HIPPI between CrayA and CrayB. If a HIPPI switch was between CrayA and CrayB, you must add the `-I` option.

Enter the following line on the slave system:

```
# /etc/vht -hCrayB-ether -i /dev/hippi0/i00 -o /dev/hippi0/o00 > -c1000 -D -P -1
```

The following text is sample output:

```
Using I-field 00000000
remsh CrayB-ether /etc/vht -hCrayB-ether -i /dev/hippi0/i00
-o /dev/hippi0/o00 -c1000 -D -P -1 -o /dev/hippi0/o00 -u 128
-t 10 -w -l 0 -n 1&
random pattern selection
HIPPI test path 0 completed pass 800
/etc/vht 0 passes 1-1000 no fatal errors
```

The following steps will run the preceding test using two connections (or windows), each connected to separate Cray Research systems, and no remote shell.

a. Enter the following command on the system that will be reading data:

```
# /etc/vht -r -i /dev/hippi0/i00 -c 1000
```

b. Enter the following command on the system that will be writing data:

```
# /etc/vht -w -o /dev/hippi0/o00 -c 1000
```

NOTE: You must initiate the read before the write so that the write will not time out. You can specify other options to the write and read. Both the server and client must have the same VHT parameters specified so each side knows how many cycles to run and which test patterns will be transmitted.

3. Configure up TCP/IP for the HIPPI interface by entering the `initif`(8) command for the HIPPI interface. The following example shows how to initialize the `hi0` interface:

```
# /etc/initif hi0
```

Test TCP/IP across the HIPPI Interface

Test TCP/IP across the HIPPI interface by completing the following steps:

1. Issue the `ping` command to the hosts listed in your `hycf` file (not the resident address); an example follows:

```
# /etc/ping CrayB-hippi
```

2. Use the `nettest` client and `nettestd` server commands to perform a TCP/IP memory-to-memory test. Use the following parameters to run on the server and client commands. Start the server side first. (The & character places the `nettest` daemon in the background.)

   Server command:

```
# /etc/nettestd -p tcp &
```

   Client command:

```
# /etc/nettest -p tcp -s 4 -b 378k -f CrayB-hippi 100 378k
```

Definitions:

`-p`	Specifies the TCP protocol.
`-s 4`	Specifies the maximum TCP window shift factor.
`-b 378k`	Specifies that 378 Kbytes of buffer should be allowed.
`-f`	Specifies that full-size read will always be issued on reading the data.
`CrayB-hippi`	Specifies the IP address.
`100 378k`	Specifies the count and size of the writes, respectively.

Output from the test is displayed on standard output on the client system.

Stop the `nettestd` daemon on the server host by entering the following command:

```
# kill -9 {PID} (Substitute the process ID
number for the PID)
```

3. The following commands can be used to further test the HIPPI interface using TCP/IP. Refer to the man page for each command for details.

   - `ping`(8)
   - `telnet`(1B)
   - `ftp`(1B)

Removed Parts Disposition

Do not dispose of removed parts locally; return the removed parts to:

Cray Research, Inc.
1000 Halbleib Road
Chippewa Falls, WI 54729
Attention: Removed Equipment Management

IR Reporting

There is a separate incident report for upgrades. Refer to *CSH # ADM-COM-9307*. Please fill one out.